Joint project

Project members: Dong Han, Rony Velastegui , Md Borhan Uddin

Spectral image

Dataset

The spectral data cubes are taken by three different cameras in the color imaging lab in UGR. PikaL is a hyperspectral linescanner type camera (400:1000 nm) with 150 spectral bands. The one with the rotatory tripod. Specim IQ is also a hyperspectral linescanner type camera (400:1000 nm) with 204 spectral bands. The one operating without a computer or rotatory tripod. SpectroCam is a multispectral filter wheel camera (400:1000 nm) with 8 spectral channels. The CMF, D65 illuminant and the reflectance of the gray flat field sample used with PikaL and SpectroCam cameras are included.

Fig. Images of camera used in the project to capture the painting

Data Preprocessing

For the Pikal and the SpectroCam cameras, the flat field correction is needed. The formula is shown below:

*Cube_corrected = (Cube_raw - Cube_black) / (Cube_flatfield - Cube_black) * gray_reflectance.*

where the *Cube_raw* is taken from painting, the *Cube_black* is taken by blocking the camera, the *Cube_flatfield* is taken by replacing the painting with the same size of gray board and the *gray_reflectance* is the reflectance values according to the wavelength 400-1000nm.

Rendering Spectral Image

Hyperspectral images can be said as cubes with two spatial dimensions (pixels) and one spectral dimension (wavelength) that provide both spatial and spectral representations of scenes. The same information can be rendered as an RGB color image. In order to render the hyperspectral image, hyperspectral reflectance image data needs to be transformed to RGB image data.

We generate an RGB representation of the scene under global illumination with 6500k. To do so, load the spectrum D65 and obtain D65 radiances. In the next step, the radiance data must now be converted into tristimulus values XYZ. Recording the array size, reshaping the size of D65 radiances properly for matrix multiplication with color matching functions CMF. Color matching functions for 2° standard observer functions are covered from 360-780nm. For converting tristimulus values XYZ to the default RGB colour space sRGB, the xyz2sRGB.m function is performed.

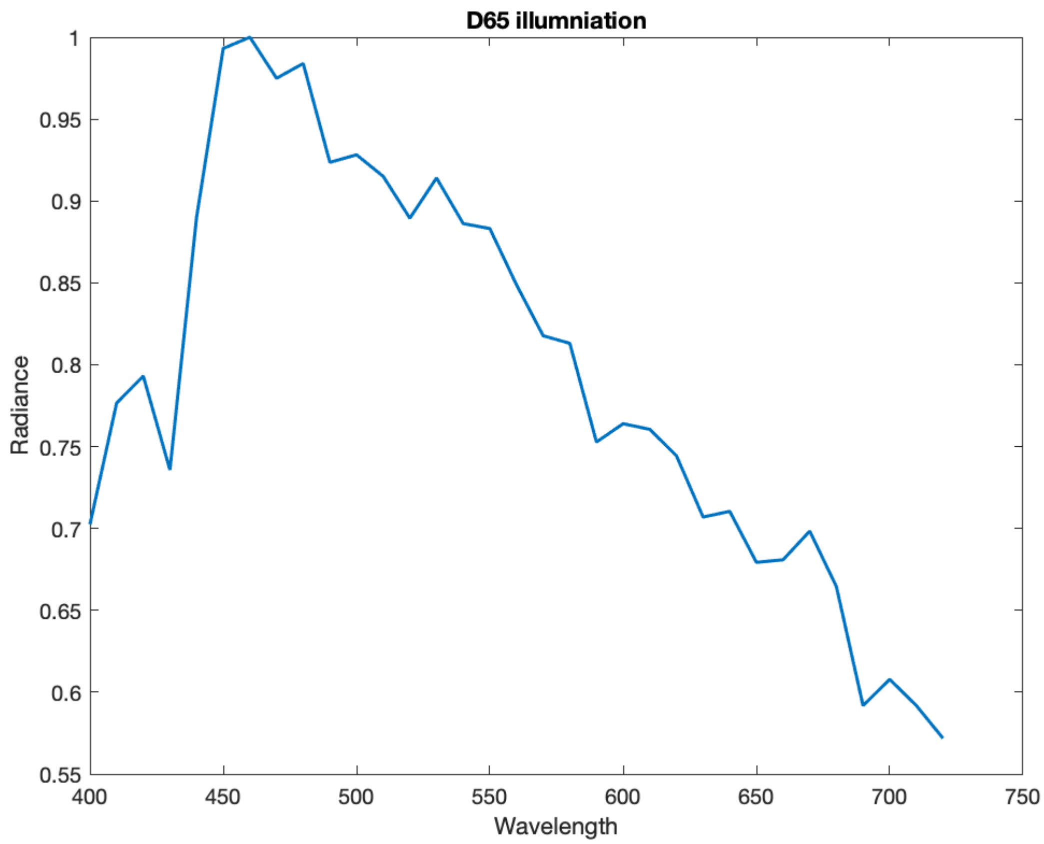

Figure 1. D65 illumination

The D65 illumination data used in the experiment is from 400-720nm.

Reconstructed results from three different cameras.

Figure 2. Displays the rendered RGB images (left, middle, right) of the hyperspectral scene from with Gamma correction 0.4, 0.5, 0.6 respectively. First row is the SpectroCam camera, the second row is the Pikal camera and the last row is the Specim IQ camera.

The CIELAB values are also computed from XYZ stimulus values. The CIELAB visualization shows below:

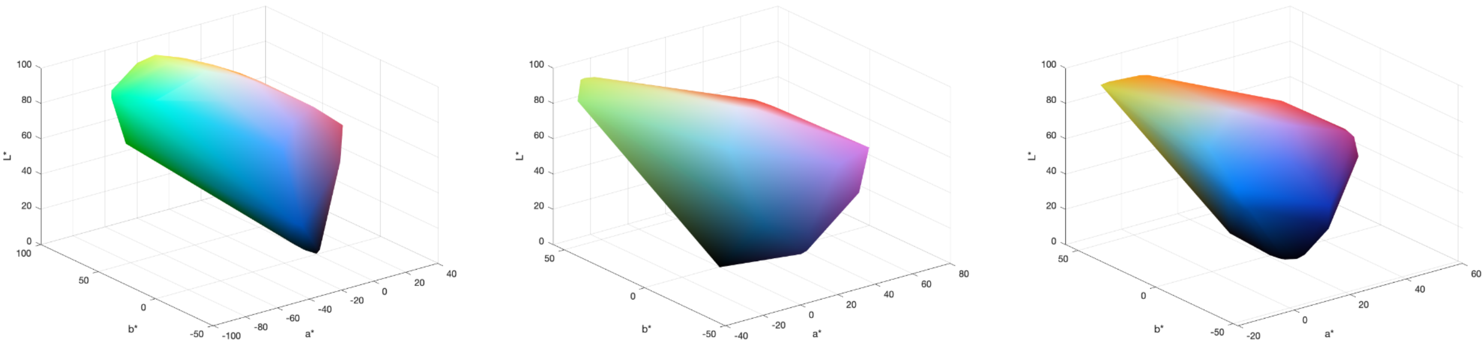

Figure 3. Visualize the CIELAB values from left to right, the SpectroCam camera, Pikal camera and the Specim IQ camera, respectively.

Gamut volumes are computed accordingly. Color Volume is the “Color Gamut” + the “Dynamic/Luminosity Range”.

Cameras	SpectroCam	Pikal	Specim IQ
Gamut volumes	355413	309269	177781
RGB Image size	2058X2456X3	900X1283X3	512X512X3

Table 1. The Gamut volumes of three devices and corresponding data size

Generally, the higher the color volume, the better the display can express a huge range of vivid and accurate colors. Obviously, the gamut volumes are different for three devices. We notice that the gamut volumes have a relationship with the original data size. In this case, the gamut volume is higher when the corresponding image has a larger size.

Hyperspectral Image Segmentation

Image segmentation is the process of subgrouping the different image regions based on some definitive criteria which is dependent on the segmentation task. It is generally done as the pre-processing step before image compression, feature extraction and pattern recognition. For segmenting different regions of the image different clustering algorithms are used such as k-means clustering. The data points are grouped together in clusters such that there is more similarity between the data points in each group and more dissimilarity between data points of other groups. There are different clustering algorithms such as k-means clustering, k medoids clustering, hierarchical clustering etc. Among all the clustering algorithms k-means is the most popular one.Through k-means clustering algorithm this clustering of data points are achieved by minimizing sum of squared Euclidean distances between features xi and their nearest cluster centers mk

The main steps of K-Means clustering can be summarized below:

1. First, randomly initial clusters k
2. For each data points randomly assign them in k different clusters
3. Then, the centers of the clusters are calculated
4. For all points distance between center of k clusters and data points are calculated
5. The points are reassigned to new cluster based on this distance
6. Again, new clusters centers are computed
7. Finally, repeat steps (4), (5) and (6) until the cluster's center converges or a stopping criterion is met. `

There are a lot of segmentation methods regarding RGB color images but only a few options for the hyperspectral image and also there is no existing built-in function in matlab. For this situation, we try to combine the Principal Component Analysis (PCA and K-Means clustering algorithm together to implement image segmentation based on hyperspectral data.A dimensionality reduction algorithm tries to reduce the dimension of the data by selecting few data points that are representative of the whole dataset. That means only those points are considered that collectively explain most of the variability in the original dataset. Although in that case sacrifice some precision of the dataset still it’s a reasonable tradeoff for simplicity of the representation and further processing of the dataset. PCA is a good strategy for hyperspectral data for this purpose. Based on PCA, we only need n numbers to specify any spectrum in nxn space. Eigenvalues and eigenvectors are the basis of PCA. Sometimes we are able to represent the whole spectrum by a limited number of eigenvectors while preserving as much information as possible.

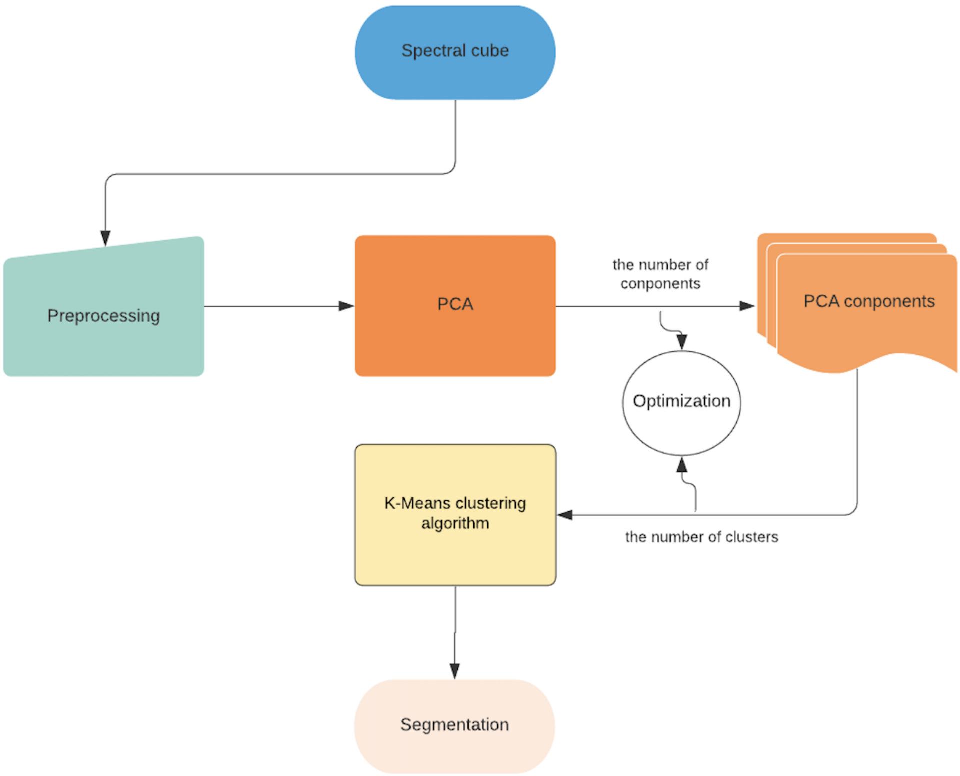

Figure 4. Workflow of spectral image segmentation

The K-mean clustering is sensitive to the outlier and the outliers of image from those three cameras are all quite big compared with the painting itself. We can see this issue clearly from **Figure 2**, the paintings are surrounded by scenes in those three cameras, they can be a potential factor to affect the segmentation accuracy. For this reason, we need to find a way to exclude the surrounding before performing the segmentation.

For the PCA component number chosen, Variance Accounted For (VAF) is a measure to represent how much variation of the original dataset is represented by the number of principal components. It gives an idea how much information of the original dataset is approximated by principal components that are selected by using PCA. In our case we are interested in performing the segmentation based on whole spectral information, so for each camera we select the maximum PCA component. The Variance Accounted For are calculated and plotting below:

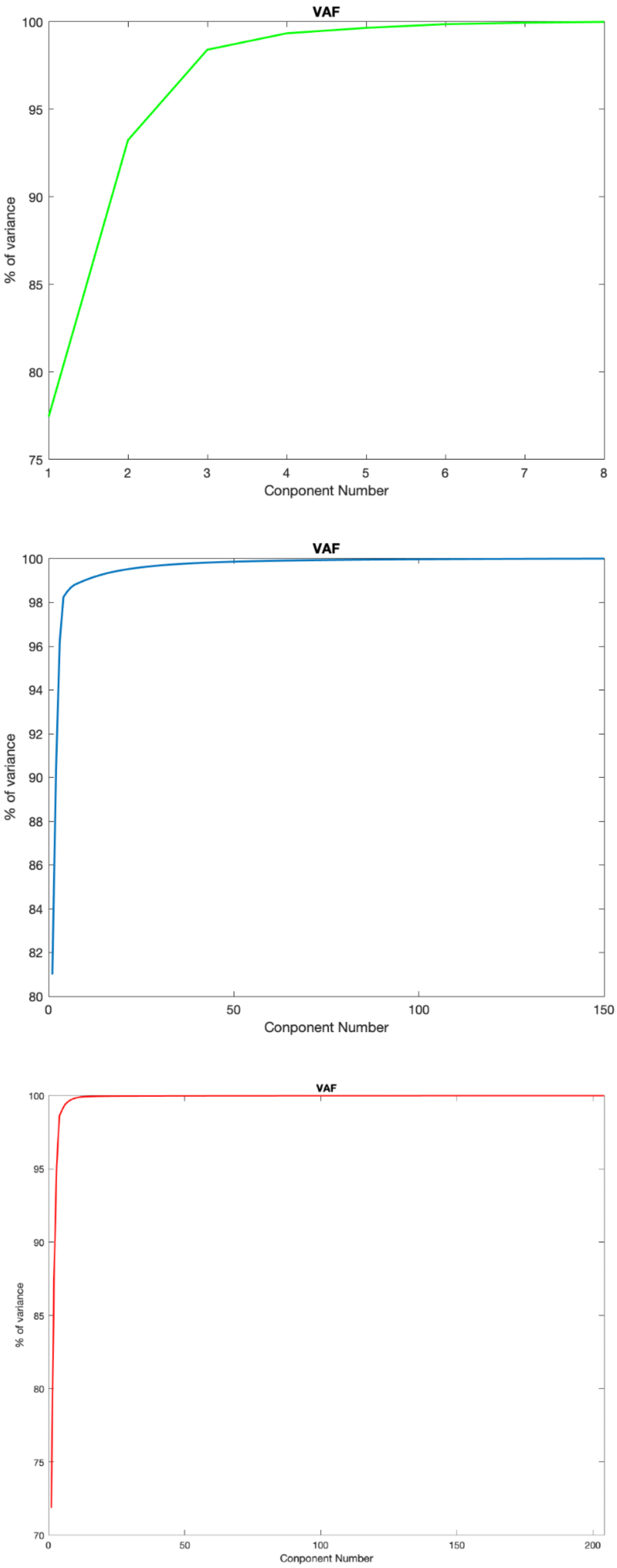

Figure 5. VAF for three spectral cube

In order to minimize errors introduced by the outlier of images. We crop the spectral cube to exclude the background of the image. The reconstructed RGB image from cropped spectral cube is shown below:

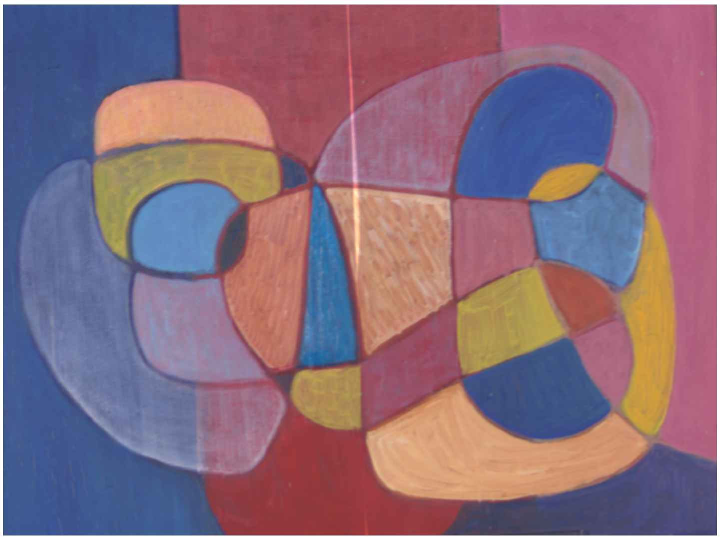

Figure 6. The reconstructed RGB image is based on the cropped spectra cube, Pikal camera camera.

The ground truth image is manually segmented by the Image Segmenter tool in matlab. After segmentation accuracy comparison with different numbers of regions, we decide to divide the image into 10 regions according to relevant colors. The same color is grouped into the same region. Therefore, we set the cluster number equal to 10 accordingly. And we also try to include whole spectral information, so the 150 PCA components are used in this case.

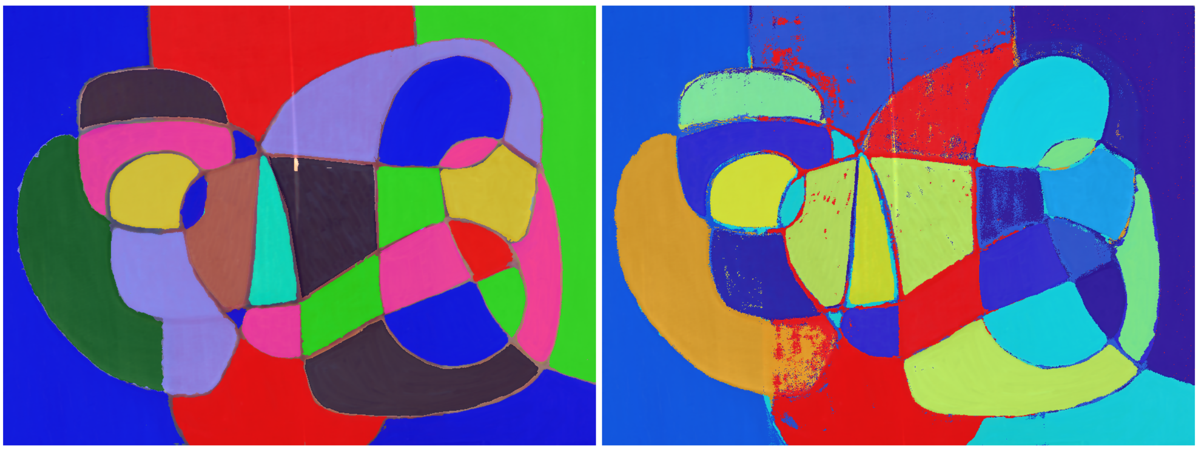

Figure 7. The ground truth RGB image and segmentation results with the number of clusters 10, the number of PCA components 150. Pikal camera.

The same procedures are applied to the other two cameras.

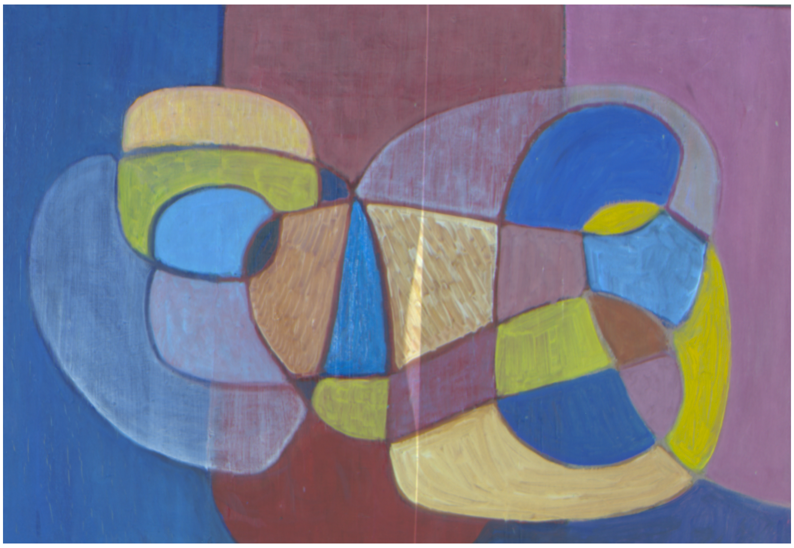

Figure 8. The reconstructed RGB image is based on the cropped spectra cube, SpectroCam camera.

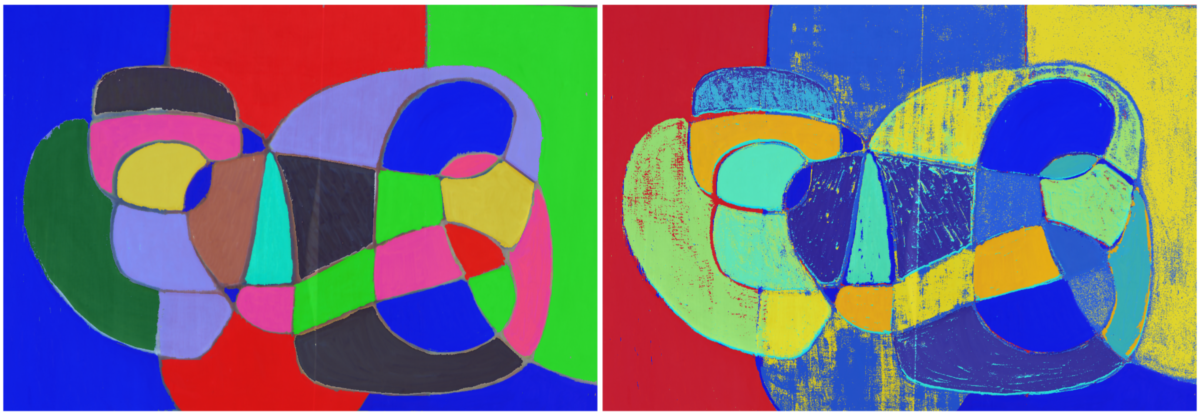

Figure 9. The ground truth RGB image and segmentation results with the number of clusters 10, the number of PCA components 8. SpectroCam camera.

Figure 10. The reconstructed RGB image is based on the cropped spectra cube, Specim IQ camera.

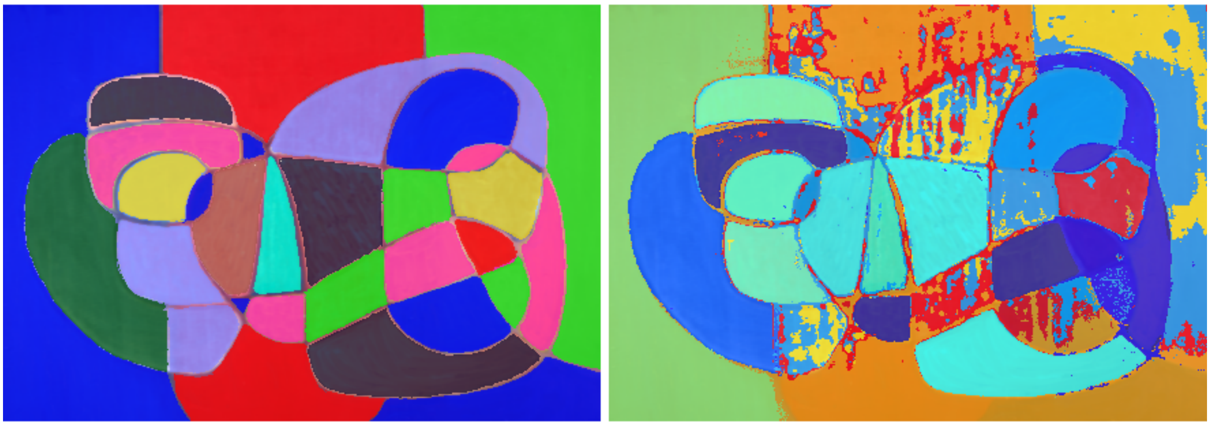

Figure 11. The ground truth RGB image and segmentation results with the number of clusters 10, the number of PCA components 204. Specim IQ camera.

From segmentation results of three devices, we can see some error parts in the segmented image. This is due to the extreme variant data that exists in the spectral cube. Since spectral data do not indicate any perceptual information, the cluster computed by K Mean clustering algorithm will not exactly group data by different color.

Segmentation Evaluation

Jaccard index is also known as Intersection over Union (IoU). It is a method to quantify the overlapped percentage between the target data and the predicted data. Jaccard index is closely related to the Dice coefficient which is often used as a loss function during the training process. Generally the IoU metric measures the number of pixels common between the target and prediction images divided by the total number of pixels present across the both images. IoU is also used for object detection in computer vision. Mathematically it can be represented as:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

The intersection (A∩B) is determined by the pixels found in both the predicted data and the ground truth data.

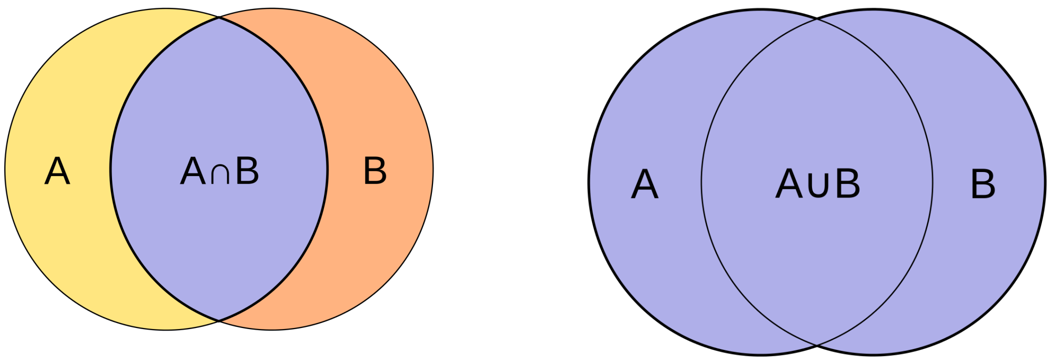

Figure 12. Intersection and Union of two sets A and B

And the union (A∪B) is simply determined by the all pixels found in either the prediction or target data. Jaccard index is widely used in ecology, genomics, medical research to efficiently measure the quality of the segmented images predicted by an algorithm with the ground truth image.

Camera	Pikal	SpectroCam	Specim IQ
Cluster 1	0	0.2533	0.1722
Cluster 2	0.0040	**0.6057**	**0.9172**
Cluster 3	0.2493	**0.5714**	0.1771
Cluster 4	**0.5804**	**0.5438**	**0.5514**

Cluster 5	0.0003	**0.5352**	0.0016
Cluster 6	0.4243	0.0257	**0.5907**
Cluster 7	**0.5560**	0.2061	**0.6140**
Cluster 8	**0.6248**	0.0109	0.0965
Cluster 9	0.4760	0.1086	0.2888
Cluster 10	**0.7932**	**0.6241**	0.4554

Table 2. The Jaccard Index of each cluster between ground truth and segmentation results.

In general, the Jaccard index is larger than 0.5 which indicates good segmentation performance. According to this criteria, the SpectroCam camera has better results than the other two devices since there are 6 clusters that exceed 0.5.

In order to see the effect of cluster number to the segmentation, we pick SpectroCam camera spectral data to do the verification. In this time, the class number increased from 10 to 21. All the spectral information is used.

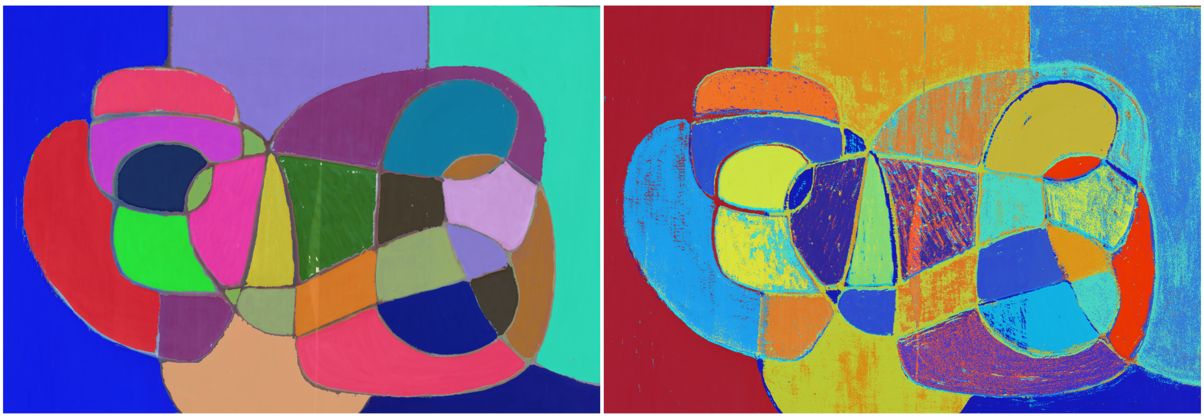

Figure 13. The ground truth RGB image and segmentation results with the number of clusters 21, the number of PCA components 8. SpectroCam camera.

0.9141	**0.7192**	0.0008	**0.8454**	0.0132	**0.708**	**0.6957**
0.4464	0.3908	0.0366	**0.7907**	0.263	0.3326	**0.568**
0.0996	0.4951	0.4693	**0.6574**	0.0017	0.1903	0.2739

Table 3. The Jaccard Index of each cluster between ground truth and segmentation results. The 21 clusters Jaccard Index values are from left to right and from top to bottom.

We can notice that the only 8 classes have Jaccard Index values which exceed 0.5. But from an average point of view (average for all the clusters), the average Jaccard Index has increased from **0.35** to **0.42.**

Moreover, if we only look at the values which exceed 0.5 in both cases.

10 clusters case	**0.6057**	**0.5714**	**0.5438**	**0.5352**	**0.6241**

21clusters case	**0.9141**	**0.7192**	**0.8454**	**0.708**	**0.6957**	**0.7907**	**0.568**	**0.6574**

We can simply calculate the average values among those successful segmented clusters. For 10 clusters case, the average is **0.58,** while **0.74** for 21 clusters case. Therefore, when the spectral information is kept the same but the cluster number increases, then accuracy of each successful segmented class will be boostered. In this case, for SpectroCam spectral data, the segmentation accuracy in successful segmented class increased **28%.**

Conclusion

In this project, we capture spectral images from the same painting by three different spectral cameras. We chromatically render RGB images from spectral data, the XYZ stimulus and CIELAB values also computed during the procedure. For the segmentation part, we use the original spectral cube to apply the K Mean clustering algorithm based on the PCA method to do the segmentation directly on spectral data. The ground truth of RGB images are made correspondly. Among the three devices, the SpectroCam gives us relatively better results according to the Jaccard index. Moreover, we find that when the number of clusters increased, the segmentation performance also increased to some extent. For future work, since spectral data do not indicate any perceptual information, the cluster computed by K Mean clustering algorithm will not exactly group data by different color. We can try to convert spectral data to other perceptual based color space first before doing the segmentation or try another segmentation algorithm.